Conjuntos universales de compuertas para computación cuántica

Laura Gatti , Jesús García López de Lacalle , Efrain Buksman , André Fonseca de Oliveira

Resumen: *En este trabajo se hace un análisis de la computación cuántica bajo el paradigma de una construcción circuital de los algoritmos. Se presenta un estudio general de los conjuntos de compuertas que permiten hacer computación cuántica general (conjuntos universales), tanto conjuntos que permiten generar cualquier compuerta cuántica como aquellos que permiten aproximar cualquier compuerta cuántica.*

Palabras clave: Quantum computation, Universal quantum gates.

Abstract: *This paper presents an analysis of quantum computation under the paradigm of quantum circuits design. A general survey of gates sets that allow overall quantum computing (universal quantum gates) is presented, for both cases of exact circuit construction and approximation by a finite set of gates.*

Keywords: Quantum computation, Universal quantum gates.

1 Introducción

La evolución de un sistema cuántico se puede representar de la siguiente forma: partiendo de un estado dado $|t_i\rangle$ se puede llegar a cualquier otro $|t_f\rangle$ mediante la aplicación de una transformación unitaria U al estado inicial $U = e^{iH(t_f - t_i)}$ donde H es el Hamiltoniano).

Esta representación es especialmente útil a la hora de hacer computación, ya que se puede ver a las transformaciones unitarias jugando el rol de las compuertas clásicas en los circuitos integrados clásicos.

Al igual que en los circuitos clásicos, la evolución de los circuitos cuánticos quedará determinada por la aplicación sucesiva de una o varias compuertas. Estas compuertas deben ser unitarias y operan sobre uno o más qubits del sistema. Si se cuenta con un sistema de múltiples qubits, (la dimensión del espacio de Hilbert es 2^n) se notará al conjunto de todas las posibles matrices unitarias de $\mathcal{H}^{2^n}$ como $Uni(2^n)$

La gran diferencia con el caso clásico, es que el conjunto $Uni(2^n)$ no es finito (en el caso clásico para un sistema de n bits, existen a lo sumo 2^n compuertas), de hecho es no númerable. Solamente inspeccionando los sistemas de un único qubit vemos que la cantidad de compuertas posibles a utilizar son infinitas:

$$Uni(2) = \begin{pmatrix} a_0 & -b_0^* \\ b_0 & a_0^* \end{pmatrix} \quad \text{con} \quad a_0, b_0 \in \mathbb{C} \text{ y } ||a_0||^2 + ||b_0||^2 = 1$$

Esta diferencia no es menor. En el modelo clásico se sabe que cualquier compuerta se puede construir utilizando únicamente una cantidad finita de compuertas NAND. Es por esto que se dice que NAND es una compuerta universal

En QC se busca tener un concepto paralelo al de compuertas universales, es decir un conjunto finito de compuertas que permitan hacer computación cuántica general. El concepto de compuertas universales en este contexto es el de un conjunto finito de compuertas que permitan aproximar con una exactitud arbitraría cualquier compuerta unitaria que se quiera. Es en este contexto que se habla de modelos discretos.

En este trabajo se presentan los principales resultados en el área tomando un orden histórico y de construcción. En la primera sección se definirá lo que es un conjunto universal y universal exacto. En la segunda sección se presenta un primer conjunto que no es finito, pero es que un conjunto universal exacto. Y finalmente se presentarán algunos de los conjuntos universales exactos finitos de mayor relevancia en el área.

2 Conjuntos universales

Una definición de **conjuntos universales** es la de un conjunto de compuertas (G) que permite aproximar con un error arbitrario cualquier compuerta unitaria que se quiera. Con mayor rigurosidad esto es pedir que el subgrupo generado por la aplicación de elementos de G sea denso en $Uni(2^n)$ para todo $n > n_0$, con n_0 fijo y típicamente pequeño.

Como caso particular, cuando permite obtener a cualquier matriz unitaria como composición de sus elementos sin error, se dice que es un **conjunto universal exacto**. Está claro que conjuntos finitos de compuertas G no podrán ser universales exactos. Ahora bien, son estos los que presentan mayor interés, ya que permitirían que, con conocidos y muy bien instrumentados bloques de compuertas, construir toda la computación cuántica.

Pensando en este objetivo es conveniente tener una definición más laxa de conjunto universal. Muchas veces es más sencillo aproximar una compuerta U, utilizando una compuerta $\tilde{U}$ de un especio de dimensión mayor mediante el uso de ancillas. Es por esto que se define un **conjunto computacionalmente universal** como un conjunto de compuertas finito G, que dado un operado $U \in Uni(2^n)$ puede ser aproximando por un operador $\tilde{U} \in Uni(2^k)$ (k>n) mediante el estado ancilla

$|\Phi\rangle \in \mathcal{H}^{2^{k-n}}$ para un vector arbitrario $|\xi\rangle \in \mathcal{H}^{2^n}$ con un error ε como

$$\left\| \widetilde{U}(|\xi\rangle \otimes |\Phi\rangle) - U|\xi\rangle \otimes |\Phi\rangle \right\| < \varepsilon ||\xi\rangle||$$

donde $\widetilde{U}$ es un producto finito de compuertas de G.

3 Conjuntos universales exactos: el $CNOT$ y las compuertas de un qubit

3.1. Compuertas controladas por un qubit

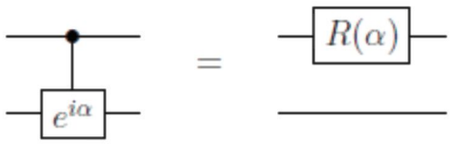
En el trabajo de 1995 de A. Barenco et al [1] se establece que el $CNOT$ y el conjunto de matrices unitarias que actúan sobre un qubit son un conjunto universal exacto. Aunque este conjunto no es finito, sienta las bases de todos los trabajos posteriores que se hicieron en el área. Por esto en la siguiente sección se mostrarán algunos resultados intermedios que permiten arribar a esta conclusión.

Como primer resultado importante se tiene el

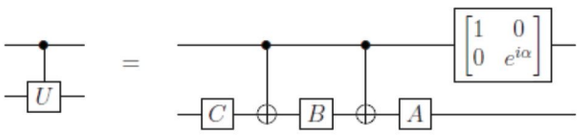
Teorema 1: Para toda $U \in \mathcal{H}^2$ existen A, B, C también pertenecientes a H^2 y $\alpha \in \mathbb{R}$ tales que: $ABC = I_d$ y $U = e^{i\alpha} AXBXC$.

La demostración consiste en descomponer U en rotaciones respecto a los ejes $\vec{z}$ e $\vec{y}$.

Este resultado en conjunto con el hecho de que



permite concluir que cualquier compuerta unitaria controlado por un qubit $\Lambda^1(U)$, Se utilizará la notación $\Lambda^k(U)$ para matrices controladas, donde k indica que se quiere controlar la matriz U mediante k controles, se obtiene entonces:

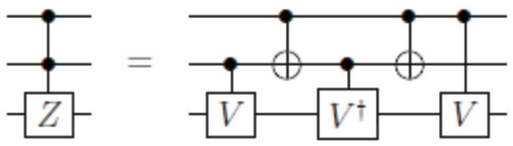


Esto es así porque si el qubit de control está en $|0\rangle$ los $CNOT$ no actúan sobre el segundo qubit, por tanto solo se aplican las compuertas $ABC = Id$. En cambio si el qubit de control es $|1\rangle$ los $CNOT$ actúan sobre el target y en definitiva se implementa $AXBXC$. Finalmente teniendo en cuenta la identidad circuital anterior, se le aplica la fase global asumiendo el control esta en $|1\rangle$. En definitiva este circuito implementa la compuerta $\Lambda^1(U)$ utilizando únicamente compuertas de un qubit y $CNOT$.

3.2. Compuertas de Toffoli generalizada

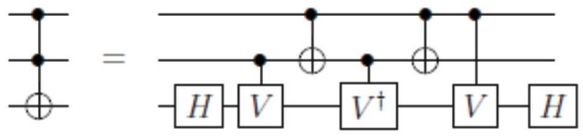
La compuerta de Toffoli: $TOFF = \Lambda^2(X)$ no es elemental, esto es que se puede descomponer como producto de matrices que actúan sobre espacios de dos o un qubits. Este resultado presentado por [2, 3] es importante por varias razones en sí mismo. Dado que la compuerta $TOFF$ es suficiente para implementar toda la lógica reversible [4] también lo serán entonces el conjunto de compuertas de un qubit y el $CNOT$. Por otro lado como se verá más adelante la compuerta de Toffoli es el bloque principal para la construcción de toda la familia de compuertas $\Lambda^k(X)$ con k $>$ 2, es decir las compuertas Toffolis generalizadas.

Desde que $V^2 = Z$ donde $V = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$ (matriz de fase) se puede descomponer la matriz Z controlados por dos qubits como:



ya que cuando los dos qubits de control son $|1\rangle$ se aplica al target $V^2 = Z$ ya que el $V^\dagger$ no se aplica (el control de $V^\dagger$ se invierte a $|0\rangle$). Si los dos qubit de control son $|0\rangle$ ninguna de las tres compuertas controladas se aplica ($0 \oplus 0 = 0$). Finalmente si uno de los qubits de control es $|0\rangle$ y el otro $|1\rangle$ se aplica al target V y $V^\dagger$, o el $V^\dagger$ y V, pero como V y $V^\dagger$ son unitarias se tiene que el $V^\dagger V = VV^\dagger = I_d$.

Dada la igualdad $X = HZH$ se puede obtener $TOFF$ como:



Por tanto la compuerta $TOFF$ se puede implementar utilizando unicamente compuertas del tipo $CNOT$ y del tipo $\Lambda^1(U)$, que ya se sabe que a su vez se pueden descomponer en compuertas de un qubit y $CNOT$.

Con este resultado el siguiente paso será construir una compuerta Toffoli generalizada (Λ^k(X)). Se verá que para esto únicamente serán necesarias compuertas del tipo Λ^{k-1}(X), $TOFF = \Lambda^2$(X) y un único qubit auxiliar. Por tanto la construcción de esta compuerta será recursiva, siendo el caso base $TOFF = \Lambda^2$(X), de la que ya se dió una construcción utilizando compuertas de un qubit y $CNOT$.

Antes de plantear la forma de construcción de las compuertas Toffolis generalizadas es importante establecer la diferencia entre el uso de una ancilla y un qubit auxiliar. Un qubit auxiliar es un qubit que se agrega al sistema. Este puede ser manipulado por una compuerta para proveer una determinada salida pero su valor a la salida debe ser el mismo que a la entrada. La ventaja que presenta este concepto de qubit auxiliar frente al de ancilla es que es muy útil en una construcción recursiva de compuertas reversibles. Dado que el valor de la salida debe ser igual al de la entrada, este puede ser reutilizado múltiples veces en vez de contar con ancillas en cascada.

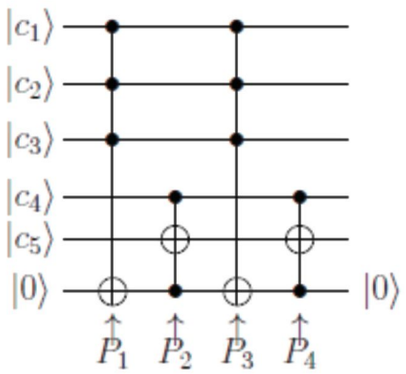
Teorema 2: Las compuertas $\Lambda^{k}(X)$ (Toffoli generelazida) se pueden construir utilizando tan solo compuertas del tipo $TOFF = \Lambda^{2}(X)$ y un qubit auxiliar.

Como es visto en [5] la idea básica será construir de manera recursiva la compuerta $\Lambda^{k}(X)$ asumiendo que se dispone $\Lambda^{k-1}(X)$ para k > 2. Para el caso k = 2 ya se dispone de $TOFF$.

Es importante recordar que el objetivo es construir una compuerta $\Lambda^{k}(X)$ que si actuando sobre un registro de qubits $\{q_1, q_2, \dots, q_k, q_{k+1}\}$ devuelve un nuevo registro de la forma $\{q_1, q_2, \dots, q_k, q_{k+1} \oplus \prod_{i=1}^{k} q_i\}$. Para realizarla se implementa la compuerta en el espacio H^{k+2} (contiene al espacio requerido más el del qubit auxiliar) utilizando las compuertas $\Lambda^{k-1}(X)$ de $\mathcal{H}^{k-1}$, la compuerta Toffoli y un qubit auxiliar. El esquema es el siguiente:

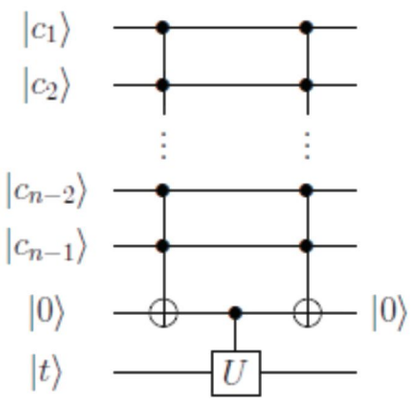
1. Esta compuerta actuará sobre un array de qubits de la forma $\{c_1, c_2, \dots, c_k, c_{k+1}, x\}$ donde c_{k+1} será el target y x el qubit auxiliar inicializado en $|0\rangle$
2. Se aplicará una compuerta $\Lambda^{k-1}(X)$ al registro con los $k-1$ primeros como controles y el qubit auxiliar $|x\rangle$ como target.
3. Luego se aplica una compuerta Toffoli sobre los últimos qubits del registro, siendo el target, el target del registro original, o sea el c_{k+1}.
4. Se repiten los pasos 2 y 3.

Se ilustra este procedimiento para el caso k = 4



Este circuito es equivalente a aplicar una compuerta $\Lambda^{4}(X)$ a los primeros $k+1$ del registro. El qubit auxiliar participa en la evolución, pero es devuelto a su valor original. En definitiva esta compuerta se puede construir a partir únicamente de compuertas Toffolis, la recursión lleva al caso base que es de hecho $\Lambda^{2}(X) = TOFF$.

Un corolario del resultado anterior es que utilizando la construcción de las Toffolis generalizadas se puede construir $\Lambda^{k}(U) \in Uni(2^{k+1})$ a partir de ellas y de las compuertas de $\Lambda^{1}(U) \in Uni(2^{2})$, que como ya se vio se pueden implementar utilizando únicamente $CNOT$ y compuertas de un qubit más un qubit auxiliar inicializado en $|0\rangle$. Para ellos basta observar el siguiente esquema:



Para ver la corrección de este esquema basta aplicar $\Lambda^{n-1}(X)$ a los primeros n−1 qubits de controles y al $|0\rangle$ se devuelve $|1\rangle$ si todos los controles son $|1\rangle$, este resultado es utilizado como control para aplicar la compuerta U al target. Finalmente se aplica de nuevo $\Lambda^{n-1}(X)$ para devolver el qubit auxiliar en el valor que fue inicializado.

3.3. Matrices unitarias cualesquiera

El objetivo final es escribir una matriz $U \in Uni(2^{n})$ cualquiera utilizando únicamente productos de matrices de las que ya se sabe que se puede obtener como producto de matrices de un qubit y $CNOT$. Para esto se hará uso de algunos resultados previos sobre matrices unitarias de los que no se presentará la demostración.

Se dice que una matriz $V_{ij} \in Uni(2^{n})$ actúa en dos niveles si dados dos vectores distintos de la base $|i\rangle$ y $|j\rangle$, estos son los únicos sobre los que V_{ij} no actúa trivialmente. Esto es:
$V_{ij}|k\rangle = |k\rangle \ \forall\, k \neq i, j$
$V_{ij}|i\rangle = a_{11}|i\rangle + a_{21}|j\rangle$ tal que $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} = \mathrm{v_{ij}}$ es una
$V_{ij}|j\rangle = a_{12}|i\rangle + a_{22}|j\rangle$
matriz unitaria.

Teorema 3: toda matriz $U \in Uni\,(2^{n})$ se puede descomponer en a lo sumo $2^{n-1}(2^{n}-1)$ matrices de dos niveles:

$$U = \prod_{\substack{i,j=0 \\ i>j}}^{2^{n}-1} V_{ij}$$

Una demostración de este teorema se puede obtener en [3]

La gran ventaja que aporta este hecho, es que las matrices de dos niveles pueden ser implementadas utilizando únicamente matrices del tipo $\Lambda^{k-1}(U) \in Uni(2^{k+1})$ y Toffolis generalizadas utilizando un sencillo esquema.

La idea básica es implementar V_{ij} a través de una compuerta del tipo $\Lambda^{n-1}(\mathrm{v}_{ij})$. Para poder hacer esto es necesario construir un camino de compuertas que mapeen el estado $|i\rangle$ en el $|j\rangle$. Si $\mathrm{a}_{n-1}, \mathrm{a}_{n-2}, \dots, \mathrm{a1}, \mathrm{a0}$ y $\mathrm{b}_{n-1}, \mathrm{b}_{n-2}, \dots, \mathrm{b1}, \mathrm{b0}$ es la representación binaria de $|i\rangle$ y $|j\rangle$ respectivamente, se puede establecer una secuencia de números $\mathrm{i} = \mathrm{R}_1, \mathrm{R}_2, \dots \mathrm{R}_k = \mathrm{j}$ que conecten $|i\rangle$ con $|j\rangle$, con la condición de que la representación binaria de dos números consecutivos disten a lo sumo en 1 bit.

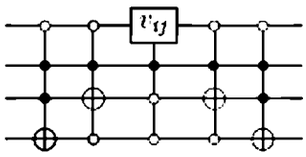
Si se toma como ejemplo i = 0111 y j = 1100 un camino posible que conecta i con j es

$$\begin{array}{rl} i = 0111 & R_1 \\ 0110 & R_2 \\ 0100 & R_3 \\ j = 1100 & R_4 \end{array}$$

Implementar una compuerta que transforme un elemento de la secuencia en el siguiente es trivial utilizando $\Lambda^{n-1}(X)$ (la única salvedad es que se tiene que permitir que los controles se apliquen cuando algunos de ellos estén en $|0\rangle$ en vez $|1\rangle$. Dado un número de la secuencia, al diferir únicamente en un qubit del siguiente, se puede obtener este último, eligiendo como target el qubit en que difieren, mientras los demás serán los controles en $|0\rangle$ o $|1\rangle$, dependiendo del resto de los valores que se quiere conectar.

Finalmente cuando se llega al número R^{k-1} de la secuencia, este difiere de j en un solo qubit y por tanto se puede implementar v_{ij} como una rotación en el subespacio del qubit en que difiere, controlado por el resto en que no. Finalmente se deshace la secuencia de compuertas que lleva $|i\rangle$ a $|j\rangle$, tomando la secuencia inversa, dejando incambiados todos los estados que no sean el $|i\rangle$ y el $|j\rangle$.

Una implementación de este circuito en el ejemplo en que se tomó i = 0111 y j = 1100 es:



Con este resultado final se termina de comprobar que cualquier compuerta $U \in Uni\,(2^n)$ se puede obtener exactamente como producto de matrices de un qubit y $CNOT$: dado que U se descompone en matrices de dos niveles y estas se implementan utilizando Toffolis generalizadas y matrices de la forma $\Lambda^k(U)$, que a su vez se estableció como estas pueden ser descompuestas en matrices de un qubit y $CNOT$ se obtiene lo deseado.

4 Conjuntos universales finitos

Luego de este resultado fue especialmente interesante empezar a buscar conjuntos de compuertas universales pero que fueran finitos. Una consecuencia inmediata del trabajo de Barenco et al [1] es que si se pudiera obtener un conjunto de compuertas finito que permitiera aproximar cualquier compuerta unitaria de un qubit con un error arbitrario este conjunto junto con el $CNOT$ automáticamente serian un conjunto universal.

Inclusive es en el propio trabajo de Barenco et al [1] que se demuestra que si se sustituye H por una compuerta R, una rotación de un ángulo múltiplo irracional de π, el conjunto que se obtiene es universal. De hecho, basta que R no preserve la base computacional para que lo sea.

La búsqueda de estos conjuntos finitos que permitan aproximar cualquier compuerta unitaria (en general llamados bases) estuvo siempre acompañada de la búsqueda de respuestas más generales sobre estos conjuntos universales. ¿Cuál será la cantidad de compuertas necesarias para aproximar una compuerta U con precisión "? ¿Estas construcciones pueden hacerse tolerante a fallos? Y finalmente ¿Que característica comparten estos conjuntos? o en otras palabras ¿Cuáles son las compuertas que brindan a la computación cuántica de su potencia sobre la computación clásica?

Un primer conjunto que se pensó podía ser universal era el conjunto compuesto por H y $CNOT$. Dado que H permite superponer estados y el $CNOT$ crear estados entrelazados parecían un conjunto prometedor. Sin embargo, Gottesman y Knill [7] probaron en 1998 que todo circuito que solo involucre a estas dos compuertas puede ser simulado eficientemente por un computador clásico.

4.1. $\{H, CNOT, S\}$ es un conjunto universal

Boykin et all en [8] al conjunto de $\{H, CNOT\}$ le agregan la compuerta $S = e^{\frac{i\pi/4}{2}Z}$. Al agregarle esta compuerta logran obtener un conjunto universal debido a que con H y S logran establecer giros irracionales respecto a ejes ortogonales con lo cual pueden aproximar cualquier compuerta de un qubit.

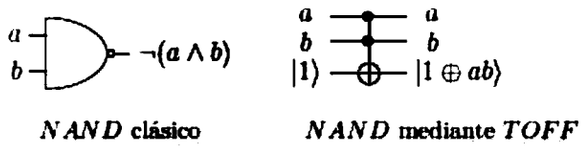
La prueba de este hecho la hacen independiente al resultado de Barenco, basándose en que **SO**(3) es localmente isomorfosmo a **SU**(2) y que cualquier giro sobre un versor $\vec{n} = (n_x, n_y, n_z)$ y ángulo θ se puede obtener como producto de giros sobre los ejes $\vec{u}$ y $\vec{v}$ y ángulos α, β y δ donde $\vec{u}$ y $\vec{v}$ son no colineales:

$$R_{\vec{n}}(\theta) = R_{\vec{u}}(\alpha) R_{\vec{v}}(\beta) R_{\vec{u}}(\delta).$$

Además los autores prueban que esta base $\{H, S, CNOT\}$ es tolerante a fallos, estableciendo que se puede construir S de manera tolerante a fallos (ya era conocido este resultado para H y $CNOT$)

4.2. $\{H, TOFF, V\}$ es un conjunto universal

Era conocido que la compuerta de Toffoli permite implementar toda la lógica clásica reversible, ya que mediante $TOFF$ se puede implementar la compuerta $NAND$:



NAND* clásico** ***NAND* mediante *TOFF

$NAND$ clásica no es reversible, sin embargo la versión cuántica de ella debe serlo, para esto se replican las dos entradas a la salida y se utiliza un qubit auxiliar para codificar la salida. Dado que $NAND$ es universal para la CC con esta compuerta se podría replicar cualquier circuito clásico que se quiera.

Shi [11] prueba que el conjunto $\{H, TOFF\}$ es un conjunto computacionalmente universal para **SO**(4). Es computacionalmente universal debido que en espacios $\mathcal{H}^{2^n}$ con $n > 3$ se requiere de un qubit auxiliar para poder

implementar permutaciones impares por lo cual para aproximar una compuerta de un espacio dado se tendrá que hacer mediantes una compuerta de un espacio de mayor dimensión.

Al ser ambas compuertas reales está claro que estas dos no alcanzan podrían ser un conjunto computacionalmente universal para $Uni(2^n)$. Pero agregando a este conjunto la compuerta $V = e^{\frac{i\pi/2}{2}Z}$ se obtiene la base buscada.

La universalidad de este conjunto podría deducirse de la universalidad del conjunto $\{\Lambda^1(V), H\}$ demostrado por Kitaev en [12]. Sin embargo es más interesante abordarlo desde el punto de vista que hizo Ahanarov en [15].

Valiéndose de que toda matriz unitaria de **SU**(2) tiene una representación real asociada en **SO**(4) deduce que a través de $\{TOFF, H, V\}$ se puede obtener un conjunto computacional universal.

Este conjunto universal permite una interpretación interesante a cerca de la computación cuántica: desde que la compuerta $TOFF$ (ancillas mediante) permite implementar toda lógica clásica reversible, parece ser que si $\{TOFF, H\}$ es computacionalmente universal es la compuerta H la que otorga la potencia a la computación cuántica. Siendo V una herramienta para llegar de **SO**(4) a **SU**(2).

4.3. Caso particular $\{H, TOFF, V, X, CNOT\}$

En [17] se presenta un modelo de computación discreto que utiliza las compuertas $\{H, TOFF, V, X, CNOT\}$. La novedad que presenta este conjunto de compuertas es que aplicado a los estados de la base computacional genera un subconjunto de estados cuánticos E con características muy especiales.

Este conjunto E para sistemas de más de 2 qubits $(n_q \geq 3)$ es denso en la esfera unitaria y permite interpretarlo como la acumulación de distintos grados de refinamiento F_k:

$$F_k = \left\{\psi \in \mathcal{H}^{2^n} : \left(\sqrt{2}\right)^k \psi \in (\mathbb{Z}[i])^{2^n} \text{ y } \left(\sqrt{2}\right)^{k-2} \psi \notin (\mathbb{Z}[i])^{2^n}\right\}$$

Donde $\mathcal{H}^{2^n}$ nota al espacio de Hilbert correspondiente a un sistema de n qubits $y\ Z[i]^{2^n}$ a un vector complejo 2^n dimensional cuya parte real e imaginaria tienen todas sus entradas enteras.

Si se define el conjunto E_k justamente como:

$$E_K = \bigcup_{k=0}^{K} F_k$$

se demuestra que que F_k es un conjunto finito de estados, con $F_k \cap F_{k'} = \emptyset$ si $k \neq k'$. Por tanto $E_{K_1} \subset E_{K_2}$ estrictamente si $k_1 < k_2$ y fundalmentalmente que $E = \lim_{K\to\infty} E_K$.

Un resultado fundamental es que E es un conjunto denso en el espacio de estados de $\mathcal{H}^{2^n}$. Por tanto este conjunto de compuertas termina definiendo un conjunto de estados que a menos de un factor de $1/\sqrt{2}^k$ tienes coeficientes enteros y permiten aproximar como un error arbitrario cualquier estado posible.

4.4. Relación entre conjuntos universales

Está claro entonces que existen varios conjuntos que son universales [10, 14, 11]. Cada uno de estos modelos permite a su vez hacer interpretaciones diferentes y complementarias acerca de la naturaleza de la información y computación cuántica.

Ahora bien, desde un punto de vista teórico surgió entonces, la necesidad de comparar estos conjuntos entre sí, por ejemplo, en cuestión de eficiencia. Algunos de estos conjuntos podrían aproximar más rápidamente en general a un circuito cualquiera.

Un teorema que da respuesta a esta cuestión es el teorema de Solovay-Kitaev [12] que refiere a la velocidad de aproximación de estos conjuntos.

Este teorema esencialmente muestra que si un conjunto finito genera un subconjunto denso en **SU**(2) entonces con este subconjunto se puede aproximar rápidamente cualquier elemento de **SU**(2) independientemente de cual sea el conjunto finito. Formalmente:

Teorema 4: (Solovay-Kitaev). Dados dos conjuntos universales cerrados bajo su inversa, entonces un circuito compuesto por t-compuertas del primer conjunto puede ser implementado con precisión ε usando un circuito de $t.poly\left(\log\frac{t}{\varepsilon}\right)$ compuertas del otro conjunto.

Demostración. Ver [12] o [13]

5 Conclusiones

En este artículo se ha presentado un resumen de algunas propuestas existentes para el diseño de circuitos para algoritmos cuánticos utilizando conjuntos de compuertas universales.

En particular se ha visto que con un conjunto finito de compuertas es posible aproximar, con precisión arbitraria, cualquier circuito cuántico.

Como caso de interés se ha presentado un conjunto de compuertas que aproximan los circuitos cuánticos trabajando con estados con coeficientes enteros (salvo un factor común de $\left(\sqrt{2}\right)^k$. Con este conjunto es posible implementar algunos algoritmos clásicos en forma exacta, como el algoritmo de búsqueda de Grover [18].

Referencias bibliográficas

[1]. A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, "Elementary gates for quantum computation," Physical Review A, vol. 52, no. 5, p. 3457, 1995.

[2]. V. V. Shende and I. L. Markov, "On the cnot-cost of toffoli gates," arXiv preprint arXiv:0803.2316, 2008.

[3]. D. P. DiVincenzo, "Two-bit gates are universal for quantum computation," Physical Review A, vol. 51, no. 2, p. 1015, 1995.

[4]. M. A. Nielsen and I. L. Chuang, Quantum computation and quantum information. Cambridge university press, 2010.
[5]. S. Xu, "Reversible logic synthesis with minimal usage of ancilla bits," arXiv preprint arXiv:1506.03777, 2015.
[6]. G. Benenti, G. Casati, and G. Strini, Principles of quantum computation and information: Volume I: Basic Concepts. World scientific, 2004.
[7]. D. Gottesman, "The heisenberg representation of quantum computers," arXiv preprint quant-ph/9807006, 1998.
[8]. P. O. Boykin, T. Mor, M. Pulver, V. Roychowdhury, and F. Vatan, "On universal and fault-tolerant quantum computing: a novel basis and a new constructive proof of universality for shor's basis," in Foundations of Computer Science, 1999. 40th Annual Symposium on. IEEE, 1999, pp. 486–494.
[9]. A. Barenco, "A universal two-bit gate for quantum computation," in Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences, vol. 449, no. 1937. The Royal Society, 1995, pp. 679–683.
[10]. C. H. Bennett and D. P. DiVincenzo, "Quantum information and computation," Nature, vol. 404, no. 6775, pp. 247–255, 2000.
[11]. Y. Shi, "Toffoli or control-not needs little help to do universal quantum computation," Tech. Rep., 2002.
[12]. A. Y. Kitaev, "Quantum computations: algorithms and error correction," Russian Mathematical Surveys, vol. 52, no. 6, pp. 1191–1249, 1997.
[13]. C. M. Dawson and M. A. Nielsen, "The solovay-kitaev algorithm," arXiv preprint quant-ph/0505030, 2005.
[14]. A. Y. Kitaev, A. Shen, and M. N. Vyalyi, Classical and quantum computation. American Mathematical Society Providence, 2002, vol. 47.
[15]. D. Aharonov, "A simple proof that toffoli and hadamard are quantum universal," arXiv preprint quant-ph/0301040, 2003. 64
[16]. C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, "Strengths and weaknesses of quantum computing," SIAM journal on Computing, vol. 26, no. 5, pp. 1510–1523, 1997.
[17]. J. Carreño and J. García López, "Conjuntos de estados para computación cuántica discreta," in XXXI Reunión Bienal de La Real Sociedad Española de Física. Real Sociedad Española de Física, 2007, pp. 291–298.
[18]. L. Gatti1, A. Fonseca de Oliveira, E. Buksman, J. García López, "Implementación del algoritmo de Grover utilizando un modelo de computación cuántico discreto," in XXXV Reunión Bienal de la Real Sociedad Española de Física. Real Sociedad Española de Física, 2015, pp. 135–136.